

REMARKS

This Amendment is intended to fully respond to the Final Office Action dated November 20, 2003. In that Office Action, claims 1-3 and 5-21 were examined, and all claims were rejected.

Reconsideration of these objections and rejections, as they might apply to the original and amended claims in view of these remarks, is respectfully requested. Claims 1, 2, 5, 7-14 and 16-21 are currently pending.

Claim Amendments

The independent claims have been amended to incorporate therein the limitation that the version-specific information relates to (and/or that the version-specific attribute is created by) a virus scanning application that is separate from the application that was used to originally create the file. These amendments do not represent new matter or new limitations but rather simply incorporate the limitations of certain dependent claims into the independent claims and address other formal matters related to the incorporation of the dependent claims. These amendments further necessitated the cancellation of dependent claims 3, 6 and 15, as well as the amendment of dependent claims 2, 14, 19 and 20. These amendments focus the pending claims on the virus scanning embodiment of the present invention and it is believed that these amendments place the application in condition for allowance. In the alternative, should the Examiner not be persuaded by the following arguments, the present amendments place the application in better form for appeal. Since the amendments do not "touch on the merits of the application," no further showing is deemed to be required under 37 C.F.R. 1.116(c) and entry of the above amendments is respectfully requested.

Claim Rejections – 35 U.S.C. § 103

Claims 1, 3-14, and 16-19 have been rejected under 35 U.S.C. § 103(a) as being unpatentable over Parker et al. (USPN 6,366,930) in view of Atkinson et al. (USPN 5,892,904). Claims 2, 15, 20, and 21 have been rejected under § 103(a) as being unpatentable over Parker et al. in view of Atkinson et al. and further in view of Hyponnen et al. (USPN 6,577,920).

Applicant respectfully traverses the § 103(a) rejections of claim 1 and the remaining independent claims since the combination of Parker and Atkinson fails to teach or suggest the invention recited in claim 1. While claim 1 has been amended to include the limitation of a virus scanning application from claim 2, the additional recitation in the Final Office Action of the Hypponen et al. patent for the § 103(a) rejection of claim 2 does not lead to a prima facie case of obviousness, as described below. Specifically, a prima facie case of obviousness first and foremost requires that the cited references must teach or suggest each of the elements of the claimed invention. MPEP §§ 706.02(j) and 2142-43. However, the combination of Parker, Atkinson and Hypponen fail to disclose or suggest the use of “version-specific attributes” relating to an independent virus scanning application as recited in claim 1 (and each of the remaining independent claims).

Parker et al.

The Office Action of June 18, 2003, relied primarily on the Parker reference in making the obviousness rejections of the independent claims, with the remaining two references being cited only for the purported disclosure of “mask information” and the disclosure of a virus scanning program, respectively. Specifically, the Office Action cites numerous sections of the Parker patent as purportedly disclosing the majority of the steps recited in claim 1 of the present application. However, these citations fail to support the rejections as alleged in the Office Action, and each citation will be addressed in detail following a short overview of the Parker patent.

Summary of Parker et al.

Initially, Parker describes a data inventory and asset management system that attempts to maximize the efficient storage of documents “by storing only a baseline version of a document offsite with all forward deltas and saving only the current version of the document on-site with all reverse deltas.” (Parker at col. 2, lines 12-15.) “An important step in this process is computing the differences between the two previous and current versions [of the file] to provide a forward delta and a reverse delta, and, then, storing the current version and the reverse delta of the changed file on-site while deleting only the last previous on-site version of the changed file, and permanently storing off-site the forward delta of the changed file and a baseline copy of each

new file.” (Parker at col. 2, lines 20-29.) “It is still a further object of the present invention to restore any requested file if it is located on-site by recovering the current version and subtracting the appropriate reverse deltas therefrom until the requested file is produced, or if the document is off-site, by recovering the baseline version and adding the appropriate forward deltas thereto until the requested file is produced.” (Parker at col. 2, lines 30-36.)

Thus, Parker describes a document management system that operates by tracking changes to a document and storing only those changes (or “deltas”) along with the current version of the document (as opposed to storing complete copies of each different version of the document) in order to preserve storage space on the computer system. In order to detect the changes made to the documents, the system of Parker conducts an inventory of all files on the computer, wherein the inventory takes note of file attributes such as the size of the file and the date and time that the file was last changed. (Parker at col. 7, line 65 to column 8, line 6.) During subsequent inventories, the Parker system identifies those files that have changed since the last inventory and generates a new “historical signature” for those files. (Parker at col. 8, lines 9-23.) Parker therefore only identifies changes to the files in furtherance of the document management invention described above (i.e., locally saving only the latest copy of a document and all the reverse “deltas” or changes that have been made to the document), and these changes are identified by reviewing standard file attributes such as the date and time that the file was last changed. Parker does not describe the creation or later use of a **version-specific attribute that pertains to the version of the document management program itself.** Simply put, Parker does not add its own version-specific attribute to a file but rather only takes note of changes to the file made by other applications (such as a word processing application).

Furthermore, Parker **teaches away** from the present invention since it relies on a thorough scan or “inventory” of every file on the hard drive in order to achieve its goal. Additionally, while the file inventory process of Parker may be used to identify changes to executable files (a known indicator of virus activity), this process requires a “scan of every byte on the hard drive file-by-file.” (Parker at col. 9, lines 25-26.) Such a heuristic approach to virus detection is far different from that described in the present application whereby an independent virus scanning application adds a “version-specific attribute” to each file scanned during an initial scan and then only rescans those files where the “version-specific information” is either

missing (due to changes to the file) or out of date due to the release of updated virus definitions. Thus, while Parker performs complete (byte-by-byte) hard-drive scans to identify files that have changed, the present invention utilizes the recited “version-specific attribute” to minimize processor overhead by only scanning those files that do not have the most current version of the “version-specific information” appended thereto. Indeed, the Background section of the present application describes some of the problems inherent with the Parker system, such as the need to check a current signature against a log in order to determine whether a file has been changed. (See Application at pp. 2-3). The Parker system employs exactly this type of “log” when using its inventory process to perform a virus checking function. See col. 9, lines 33-37: “Use signatures of changed files to identify files that have changed by comparing to the previous signature in the File Clerk database. If any executable files have changed, the condition is identified as file corruption and a possible virus situation.” Thus, the system of Parker does not utilize a simple file attribute to determine whether a file needs to be scanned for viruses, but rather compares a current version of the file to a prior version to see if the file has changed. Only then does Parker identify the file as “a possible virus situation.” Col. 9, line 37.

In sum, Parker follows precisely the same type of procedure that is described in the Background section of the present application and which is rendered unnecessary by the use of the currently recited “version-specific attribute.” Parker thus does not teach or suggest (and actually teaches away from) the use of “version-specific attributes” that are added to each file by the virus checking program in order to streamline future virus checks and reduce processor overhead. To the extent that the Examiner considers the signatures of changed files to contain information regarding “versions,” such information relates to different versions of the file itself and not to the version of a separate virus scanning application that must determine whether to scan the file.

Office Action Citations to Parker et al.

The Office Action of June 18, 2003, contains a number of specific citations to Parker (starting at page 2 of the Action) that are addressed below. Applicant respectfully traverses that these citations either disclose or suggest the recited claim limitations for which they are applied by the Examiner.

For instance, column 9, lines 60-65 of Parker does not describe a method of providing **version specific information** as alleged in the Office Action. Rather, the cited portion of Parker relates to the creation of a current version of a document from a stored (archived) baseline version together with stored “forward deltas” so that the current version of the document can be recreated. There is no mention of a file attribute that relates to version-specific information of a separate application (i.e., a virus scanning application) that was used to create the attribute. Indeed, there is no mention that the document management program described in Parker creates any type of file attribute.

Column 9, line 65 to column 10, line 5 of Parker does not describe receiving a request to create a specific attribute associated with the file as alleged in the Office Action. Rather, this portion of Parker describes the retrieval of an earlier (archived) version of a document by retrieving the latest stored version and then subtracting “the required reverse deltas from the latest full version” to achieve the desired archived version. This process does not rely on the use of a **version-specific information** such as the date of a virus definition file. Rather, it only relies on the record of changes made to the file (e.g., a word processing document) so that a desired earlier version can be recreated from the current version that is stored locally on the computer system.

Column 10, lines 5-11 of Parker does not describe maintaining the version-specific attribute to reflect relevant updates to the file by invalidating the version-specific information in response to a predetermined event. Rather, this portion of Parker describes retrieving an “archived” document that has been recreated as described above. Because Parker does not maintain “version-specific attributes,” Parker does not describe invalidating the version-specific information in response to events such as changing the contents of a file. Indeed, Parker teaches away from this invention by specifically keeping track of each change to a file in lieu of

checking to see if a “version-specific attribute” has been invalidated due to such a change. This is an important distinction since Parker must keep track of every change to a file (at the expense of processor overhead) in order to achieve its desired goal of reducing the storage space required for archiving the different versions of a file. On the other hand, the present invention utilizes the “version-specific attribute” to minimize processor overhead by only requiring the virus scanning application to scan those files that have invalid version-specific information. Thus, to the extent that Parker relates at all to virus scanning applications, Parker operates in a conventional manner as described in the Background section of the present application.

Column 10, lines 12-17 of Parker does not describe receiving a request by a virus scanning application to evaluate the version-specific attribute. Rather, the cited portion of Parker again simply describes the process of retrieving a baseline version of the document from off-site storage and then adding “forward deltas” as necessary to recreate the desired archive version of the file. The system of Parker does not utilize or evaluate any “version-specific attribute” of the file in order to achieve this reconstruction.

Similarly, column 10, lines 38-43 of Parker does not describe providing version-specific information to a virus scanning application in response to the request to evaluate the “version-specific attribute.” Rather, this portion of the Parker specification merely describes transmitting the completed (reconstructed) file back to the user from the off-site storage location.

Thus, far from describing the present invention’s use of “version-specific attributes,” the cited portions of Parker only describe the reconstruction of prior versions of a document based on current (or stored baseline) versions of the document together with reverse (or forward) “deltas.” Claim 1 of the present application clearly distinguishes the Parker document management system, and the brief discussion of using the file inventory system of Parker to help identify corrupted files (column 9, line 23-37) does not teach or suggest the current invention. Rather, this portion of Parker describes a system similar to those described in the Background section of the present application that rely on comparing files to logs of past reviews (rather than assigning an attribute to the file and then checking to see if the information stored within the attribute has been invalidated).

Atkinson et al.

The Office Action of June 18, 2003, combines the above citations to Parker with two citations to Atkinson as purportedly disclosing the use of meta information and “mask” information within the “version-specific attributes” recited in claim 1. However, Atkinson fails to disclose or suggest the use of “version-specific attributes” that are added by a virus scanning application (and later used by the same virus scanning application to determine whether the file needs to be scanned against a current version of the virus definition file). Rather, Atkinson discloses the use of authenticating signatures provided by a software publisher to verify that the software has not been altered during transmission to an end user. “The code signing method assures the recipient of the identity of the publisher as the source of file (i.e., its authenticity) and that the file has not been modified after being transmitted by the publisher (i.e., the integrity of the file). As a result, the code signing method allows an executable file to be transmitted over open computer networks like the Internet with increased certainty in the identity of the source of the file and minimized risk of contracting a computer virus or other malicious executable computer files.” Atkinson at col. 2, lines 44-52.

Thus, Atkinson does not describe or suggest the creation and later use of a “version-specific attribute” for files on a computer system. Rather, Atkinson describes the digital signature of a file so that the file may be transmitted to the computer system and the user may be confident that the file was actually sent by the authorized distributor and was not corrupted during the transfer. Once the file arrives, the digital signature is reviewed by the user to ensure authenticity. However, the digital signature does not include “mask information” relating to predetermined events that will invalidate the signature. Rather, a public key is applied to the signature to determine whether the signature is valid. If the digital certificate cannot be properly decrypted by the public key, the user is notified that the publisher’s signature is invalid. (Atkinson at col. 7, lines 56-67.) Indeed, the use of a key to determine the validity of the file actually **teaches away** from the use of “mask information” to define a variety of different invalidating events.

Thus, Atkinson does not provide the missing disclosure of Parker (i.e., the generation of a “version-specific attribute” by a virus scanning application), nor does Atkinson disclose the

inclusion of “mask information” within such an attribute. Reconsideration of the §103 rejections of the independent claims over the combination of Parker and Atkinson is therefore respectfully requested.

Hypponen et al.

As amended, claim 1 and the remaining independent claims now recite that the application responsible for the version-specific attributes constitutes a virus scanning application. This limitation was formerly found in dependent claims such as claims 2, 15, 20 and 21 that were rejected under § 103(a) in light of Parker and Atkinson as described above, and further in view of Hypponen et al. While Hypponen et al. describes a virus checking application, this patent does not supply any of the above missing elements from the combination of the primary Parker and Atkinson references. Specifically, Hypponen et al. does not teach or suggest utilizing a “version-specific attribute” to help streamline the virus checking process. Rather, Hypponen et al. describes the use of multiple databases (one of known viruses and others containing known “good” macros) to help increase the accuracy of the virus scan. Hypponen thus teaches away from the present invention by teaching the prior art process of checking each file against these databases (see FIG. 2). The present invention represents an improvement over Hypponen et al. (and other similar virus scanning programs) by adding a “version-specific attribute” to each file so that future virus scans will only be performed on those files that have been changed since the last scan or on those files that have not been scanned with the latest virus definitions.

No Prima Facie Case of Obviousness For The Independent Claims

The combination of Parker, Atkinson and Hypponen fails to disclose or suggest all of the steps of claim 1 and the remaining independent claims. Indeed, none of these references, alone or in combination, discloses or suggests the use of a “version-specific attribute” that is added to a file by a virus scanning application to help reduce processor overhead when performing a virus scan. As noted above, Parker describes a method of tracking changes to a file so that different versions of a file can be retrieved by a user without the need to actually store each separate version. Specifically, Parker teaches recreating a prior version of a document by taking the current version stored locally on the system and then applying reverse “deltas” as necessary to achieve the prior version. Such a system does not rely on the use of “version-specific attributes”

stored with each file (such as the virus definition file that was last used when the file was scanned for viruses). None of the extensive citations to the Parker patent in the Office Action relate to the currently claimed invention, but rather relate only to the recreation of prior “versions” of a document. This is a far different invention than that claimed in the present application and these citations to the Parker patent cannot form the basis of an obviousness rejection.

Furthermore, neither the Atkinson patent nor the Hypponen patent cure any of the deficiencies of the Parker patent. Indeed, Atkinson was cited solely for the purported provision of “mask” information being included with the “version-specific attribute.” However, not only do the digital signatures of Atkinson fail to describe or teach the claimed “version-specific attributes,” but the use of a public key to determine the validity of the signature does not comprise “mask” information that is included with the file attribute itself. Furthermore, Hypponen teaches away from the present invention by disclosing a virus scanning application that checks each file against multiple databases. This conventional virus scanning approach was described in the Background section of the present application and is easily distinguishable from the pending claims that recite the use of a specific file attribute appended to each file to help reduce the processing overhead associated with a virus scan (i.e., by skipping the scans of those files that have not been altered and/or that have already been scanned with the latest virus definitions). Thus, in light of the above arguments, reconsideration of the § 103(a) rejections of the independent claims is respectfully requested.

The Remaining Dependent Claims Are Patentable

Each of the pending dependent claims are believed to be allowable in light of the above arguments. Indeed, several of the dependent claims have been canceled or amended in light of the amendments to the independent claims described above.

CONCLUSION

As originally filed, the present application included 21 claims, 4 of which were independent. As amended, the present application includes 17 claims, 4 of which are independent. Accordingly, it is believed that no other fees are due with this Amendment and

Response. However, the Commissioner is hereby authorized to charge any deficiencies or credit any overpayment with respect to this patent application to deposit account number 13-2725.

In light of the above remarks, it is believed that the application is now in condition for allowance, and such action is respectfully requested. Should any additional issues need to be resolved, the Examiner is requested to telephone the undersigned to attempt to resolve those issues.

Respectfully submitted,

Dated: 1/20/04



T.B. Stull

Timothy B. Stull, #42,137
MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, MN 55402-0903
303.357.1648